International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18 (2026) pp. 1-17

https://doi.org/10.70917/ijcisim-2026-1111
Article

Research on Personalized Learning Path Planning and Career Development Linkage Based on Reinforcement Learning

Qing Xiong [1,*] and AZLAN BIN ABDUL LATIB [1]

[1] Faculty of Social Sciences and Humanities, School of Education, University of Technology Malaysia, Johor Bahru, Malaysia
* Correspondence author: xiongqing139@163.com

Abstract: In the context of the rapid development of online education, merely providing learners with a large amount of learning resources is insufficient to meet their personalized learning needs. This paper proposes an adaptive learning path recommendation model based on reinforcement learning, exploring its potential applications in enhancing learning outcomes and connecting with career development. The knowledge tracking model (SKT) based on multi-head self-attention mechanism is applied to model the learners' constantly changing knowledge levels. The cognitive navigation algorithm is introduced to filter the candidate learning item set, enhancing the logic of the learning path and reducing the search space of the strategy function during the recommendation process. The recommendation is carried out using the reinforcement learning algorithm, incorporating the degree of change in the learners' knowledge levels into the calculation of the reward function, to evaluate the quality of the recommended items more precisely. Experimental results show that the RL4ALPR model proposed in this paper outperforms the comparison algorithms in both single-peak and multi-peak functions. The learning paths recommended based on this method are used by students, and the target LO correct rate is the highest, with a value of 0.84. These results verify the effectiveness of the RL4ALPR model in improving students' learning outcomes. It provides a feasible technical path for achieving intelligent integration of personalized learning and career development.

Keywords: Multi-head Self-Attention Mechanism; Knowledge Tracking; Cognitive Navigation Algorithm; RL4ALPR Model; Personalized Learning Path Planning

1. Introduction

With the support of many new technologies, such as mobile Internet, big data, artificial intelligence, cloud computing, etc., the education model, teaching methods, learning styles, etc., are undergoing profound changes, prompting China's education to move in the direction of wisdom and intelligence [1]. The concepts of “paying attention to the individual differences of learners” and “providing suitable education for each learner” have gradually gained consensus in the society, and personalized education and personalized learning have become the main way to solve the contradiction between supply and demand of education in China's current society [2]. In the digital era, learning resources are greatly enriched, and how to use intelligent technology to provide learners with customized learning paths is a hot topic in the current research field of personalized learning. The problem of personalized learning path recommendation can be defined as, based on the differences in learners' learning ability, knowledge background, learning interest, achievement goals, etc., through intelligent technology to customize a learning path for learners that conforms to the laws of education and can achieve the learning standards, and at the same time to achieve the learning status detection of learners [3-4].

Since the 1960s, researchers have explored a variety of algorithms for personalized learning path planning, including algorithms based on graph theory, traditional machine learning algorithms, and

deep learning and reinforcement learning algorithms that have been widely used in recent years [5]. Literature [6] proposes a learning path recommendation method for knowledge construction and learning performance analysis, which integrates learners' static and dynamic characteristics to generate personalized learning paths, dynamically adjusts the difficulty of resources based on students' real-time learning performance, and pushes learning content in combination with learning preferences and ability adaptation, and at the same time predicts each learner's learning duration and expected score. Literature [7] introduces the improved Dijkstra algorithm into the domain model based on the graph structure, which takes into account the learner's existing knowledge, the weight of the course knowledge points and the influence of the association between the knowledge points, predicts the probability of success of the learning of each knowledge point, and then recommends the optimal learning path for the students based on the probability. Literature [8] proposes the Nestor hybrid artificial intelligence algorithm, which integrates qualitative experience and quantitative data, and integrates multiple learning theory dimensions such as learning styles, learning strategies, personality traits and learning preferences. Literature [9] analyzes three kinds of knowledge point topic sequences in personalized adaptive e-learning systems: teacher-set, learner-directed, and optimal planning. Literature [10] proposes a multi-algorithm collaborative personalized learning path recommendation model, which constructs a learner model from four aspects: cognitive level, learning ability, learning style, and learning intensity, and uses association rule algorithms to mine the association of knowledge points, plan the learning sequence, and then matches each knowledge point with a high degree of fitness of the personalized learning resources through the swarm intelligence algorithm.

Personalized learning path recommendation is essentially a sequence decision-making problem, and reinforcement learning, as an effective sequence decision-making method, can continuously optimize the learning path decision-making in a dynamic interactive environment. Therefore, reinforcement learning shows great potential in this field, and has achieved relatively excellent recommendation results, and gradually developed into the mainstream algorithm in this field. Literature [11] combines Markov Decision Process MDP and Reinforcement Learning RL to construct an intelligent online learning framework, and applies Q-learning and other algorithms to sequential learning path recommendation, with the help of MDP to dynamically adjust the recommendation strategy according to the learner's feedback, and to adapt to new learning activities and path planning. Literature [12] proposes an adaptive learning path navigation ALPN system for providing highly personalized adaptive learning paths for online learning platforms, which integrates the attention knowledge tracking AKT model to assess the learner's knowledge state, and optimizes the learning resource recommendation with the proposed entropy-enhanced proximal policy optimization EPPO algorithm. Literature [13] points out that reinforcement learning is good at portraying the complex associations between course activities, learner behaviors and learning outcomes, and combined with learning path recommendation, it can provide a broad application space for personalized education, and for this reason, the proposed dual-constraint deep Q-network DCQN offline reinforcement learning method. Literature [14] builds a learning path recommendation system based on graph reinforcement learning: from the learner query and knowledge base to determine the starting point and end point of the learning path, learning resources and their associations are constructed as a graph structure, with the nodes representing the learning objects, and the edges characterizing the degree of association; and Q learning algorithms are introduced into the corpus search space of the resource to match the user with the optimal learning path that maximizes the use of their own existing knowledge. Literature [15] proposes an efficient personalized learning path recommendation algorithm, which firstly integrates the learning behavior module into the dynamic key-value memory network DKVMN to realize the accurate tracking of students' knowledge status, and then uses the model to simulate the virtual students and combines it with the reinforcement learning to train the learning path planning strategy.

For individuals, the core crux behind blindly taking certificates and brushing up on classes, and finally finding that they have lost touch with their preferred positions, lies in the disconnection between learning path planning and career development. Literature [16] suggests that students need to bind their academic planning with career goals from the beginning of their schooling, and when developing learning paths, they not only need to complete the requirements of the curriculum, but also need to plan for internships, apprenticeships, comprehensive practice, and other academic-related practices and subsequent vocational qualification certification, so that the entire learning path will always be in service of the needs of career development. Literature [17] puts forward effective strategies at different levels, the essence of which is to integrate career development learning into the whole learning path of learners through curriculum design, interdisciplinary collaboration, etc., to realize the in-depth connection between learning planning and career development, to help learners improve their employability and clarify their career direction, and to make the learning path always focus on the goal of career development. Literature [18] believes that personal development plan (PDP) as an important

learning and development tool, its core value is to build a bridge between learning path and career development, in PDP practice, the learning path planning and career development needs in depth binding, not only relying on the PDP to help learners to clarify the direction of learning, improve core competencies, but also through the PDP to link the informal learning and career development goals, letting the learning path always centers on career growth. Literature [19] argues that career development learning (CDL) can help students reflect on and plan their future careers and improve their employability based on extracurricular activities, and is an important carrier for bridging learning paths and career development, and that the normalization of vocational literacy into teaching and learning planning can realize the deep connection between learning paths, literacy development and career development.

This paper proposes a personalized learning path recommendation method based on reinforcement learning. This method uses a knowledge tracking model (SKT) based on multi-head self-attention mechanism to dynamically model the changes in learners' knowledge levels, more accurately capturing the long-distance dependency relationships between learners' knowledge states. The cognitive navigation algorithm is introduced to complete the screening of the learning item set, enhancing the logical coherence of the recommended path. The changes in learners' knowledge levels are introduced into the calculation process of the reward function, so as to evaluate the contribution of each recommended item to the learning effect more finely, thereby guiding the model to generate more excellent personalized learning paths. To verify the effectiveness of the model, single-peak and multi-peak functions are applied to validate the performance of the model, with the target correct rate of the LO question as the core evaluation index, and the RL4ALPR model and other types of recommendation methods are compared.

2. Adaptive learning path recommendation model based on reinforcement learning

2.1. Reinforcement Learning Algorithm

In reinforcement learning, there are two interacting entities: the agent and the environment.

The basic elements of reinforcement learning include:

State: The state space is represented as S, and the state of the environment at time t is represented as $s_t \in \mathrm{S}$.

Action: The action space is represented as A, and the action executed at time t is $a_t \in \mathrm{A}$.

State transition probability $p\left(s'|s,a\right)$: It describes the dynamic laws of the environment. However, in general, the dynamics of complex environments are difficult to quantify, so the state transition probability is usually invisible.

Reward: It is a scalar function that reflects the quality of the agent's behavior or the environment state reached, represented as $r(s,a,s')$.

Policy: It is a function represented as $\pi\left(a|s\right)$, which guides the agent to select action a when in state s. It can be divided into deterministic policies and stochastic policies.

The prerequisite for applying reinforcement learning algorithms to real-world scenarios is to model the problem to be solved first. Reinforcement learning uses Markov Decision Processes (MDPs) to model the environment to seek the optimal solution for the problem to be solved.

An MDP is composed of a five-tuple $(\mathrm{S},\mathrm{A},\mathrm{P},\mathrm{R},\gamma)$. Here, S represents the state space; A represents the action space; P is the state transition probability matrix, where $\mathrm{P}_{ss'}^{a} = p(s'|s,a)$; R is the reward function, where $\mathrm{R}_{ss'}^{a} = r(s,a,s')$, and $\gamma \in [0,1]$ is the discount factor. The interaction process between the agent and the environment can be regarded as an MDP process. Initially, the agent observes the environment's state as s_0, then executes a corresponding action a_0, and the environment's state changes to s_1 and returns a reward r_1 to the agent. Then, the agent executes a_1, the environment becomes s_2, and the agent receives the reward r_2 returned by the environment. This iterative interaction continues until the final state is reached. Such a process is called an episode and can be represented in sequence form as $s_0,a_0,s_1,r_1,...,s_{t-1},a_{t-1},s_t,r_t,...$. The MDP satisfies the Markov property, meaning that the state at the next time step, s_{t+1}, is only dependent on the current state s_t and the action a_t, that is, $p\left(s_{t+1}|s_0,a_0,...,s_t,a_t\right) = p\left(s_{t+1}|s_t,a_t\right)$, where $p\left(s_{t+1}|s_t,a_t\right)$ represents the state transition probability.

The goal of reinforcement learning is to learn a policy $\pi_\theta(a|s)$ that maximizes the expected total reward G, where the total reward is the accumulated discounted rewards during a certain interaction between the agent and the environment. The cumulative return for an episode τ is expressed as $G(\tau)=\sum_{t=0}^{T-1}\gamma^t r_{t+1}$, where the discount factor is used to reduce the weight of future rewards. When γ approaches 0, it indicates that the agent values short-term rewards more; conversely, when it approaches 1, it means that long-term rewards are more important for the agent. The objective function of reinforcement learning can be expressed as $J(\theta)=\mathbb{E}_{\tau\sim p_\theta(\tau)}[G(\tau)]$, where θ is the parameter of the policy function.

To estimate the expected return of the policy π, the basic algorithms of reinforcement learning set up state value functions and state-action value functions, and use the Bellman equation to transform them into an iterative form. The state value function $V(s)$ represents the expected return that the agent obtains from state s based on the current policy. $V^\pi(s)=\mathbb{E}_{a\sim\pi(a|s)}\mathbb{E}_{s'\sim p(s'|s,a)}[r(s,a,s')+\gamma V^\pi(s')]$. Similarly, the state-action value function $Q(s,a)$ is introduced to represent the expected return that can be obtained by performing action a in the current state s while following the policy, $Q^\pi(s,a)=\mathbb{E}_{s'\sim p(s'|s,a)}[r(s,a,s')+\gamma Q^\pi(s',a')]$. The value function can be regarded as an evaluation of the policy. An optimal policy can be obtained by maximizing the state-action value function, that is, the optimal policy $\pi^*(a|s)=\arg\max_a Q^*(s,a)$. According to the Bellman equation, the optimal state-action value function $Q^*(s,a)$ can be calculated using the value iteration method:

$$Q^*(s,a)=\mathbb{E}_{s'\sim p(s'|s,a)}\left[r(s,a,s')+\gamma\max_{a'}Q^*(s',a')\right] \tag{1}$$

The temporal difference learning (TD learning) method is a classic approach in reinforcement learning used to solve the optimal policy of a Markov decision process. It combines dynamic programming and Monte Carlo methods. By simulating a trajectory, the agent evaluates the value of the state before taking an action using the Bellman equation, which is the value function.

The DQN algorithm is one of the typical offline learning methods. It is an advanced version of the Q-learning method. The goal of Q-learning is to obtain the optimal value $Q^*(s,a)$. In Q-learning, the action a_t generated by interacting with the environment at time t is produced by an exploratory $\varepsilon-greedy$ behavior strategy, while the action a_{t+1} used to update the Q value at time $t+1$ is generated by a completely greedy target strategy. The update method of the Q value is given by formula (2):

$$Q(s,a)\leftarrow Q(s,a)+\alpha\left[r+\gamma\max_{a'}Q(s',a')-Q(s,a)\right] \tag{2}$$

In order to calculate the state-action value function in the continuous state space and action space, neural networks are introduced. A function $Q(s,a;\theta)\approx Q(s,a)$ is used to approximately calculate it, which is called value function approximation. Here, θ represents the parameters and $Q(s,a;\theta)$ is called the Q network. The stochastic gradient descent (SGD) method is adopted to optimize the objective function $\ell(\theta)$, and the loss function is as follows:

$$\ell(\theta)=\left[r+\gamma\max_{a'}Q(s',a';\theta)-Q(s,a;\theta)\right]^2 \tag{3}$$

Among them, s' and a' represent the vector representations of the next state and the next action respectively. Additionally, based on the characteristics of action a, it is possible to decide whether to use the action as an input.

The problem of sparse rewards is a common challenge faced by reinforcement learning agents in real-world tasks. The solutions to the problem of sparse rewards can be mainly categorized into two types: one is reward shaping, and the other is curriculum learning.

2.2. Recommendation Algorithm Based on Reinforcement Learning

The recommendation system environment is described as a Markov Decision Process (MDP), and it is solved using reinforcement learning methods. In addition to the selection of the recommendation

algorithm, constructing the recommendation system environment as an MDP is also very important. It can be said that the suitability of an MDP directly affects the learning efficiency and performance of the agent. In the recommendation system environment, the setting of MDP(S,A,P,R,γ) is usually as follows:

State space S : Represented by the user's displayed or implicit features. For example, the user's historical records, the representation vector of the user's preference degree obtained through user modeling, etc.

Action space A : The set of recommended items. The action a to be executed at a certain time is a project recommended by the recommendation algorithm to the corresponding user.

State transition probability P : In the recommendation system environment, the state transition probability is invisible and is obtained through the interaction between the agent and the environment.

Reward function R : Depending on the user's feedback, it is determined by specific tasks. It can be designed based on explicit results such as clicks, skips, revisit, whether to purchase or the duration of stay, or implicit results such as user stickiness.

The recommendation process of the recommendation system based on reinforcement learning is shown in Figure 1. The environment consists of four parts: users, user history logs, user feature representations, and reward calculation modules. Users give corresponding feedback behaviors f_t (such as clicking, skipping, and purchasing) based on the recommended items a_t by the intelligent agent, and leave the user's history records. Through user modeling, the user's history logs can be further represented as the user's feature representations, which can reflect implicit information such as user preferences or user intentions to represent the user's state s_t. Through the reward calculation function, the immediate reward r_t for the intelligent agent to learn is calculated based on the user's current feedback. The role of the intelligent agent is to recommend suitable items to the user based on the user's state. Specifically, the intelligent agent recommends a project a_t to the user based on the current user state and the current strategy. The user will give a corresponding feedback f_t based on this project and form the user's history record. Based on the user's feedback, the immediate reward is calculated through the reward calculation function, and the user's next state s_{t+1} is obtained through user modeling. The intelligent agent obtains the next state and the immediate reward and then conducts the next round of recommendation.

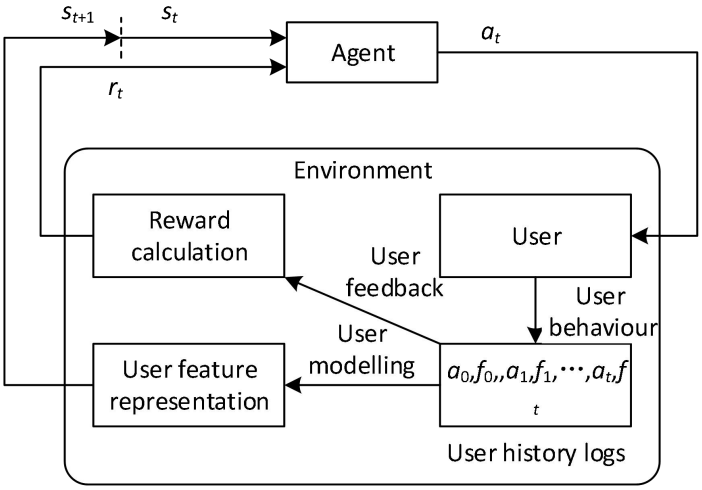

Figure 1. Recommended process based on intensive learning

2.3. Problem Definition

The definition of adaptive learning path recommendation is as follows: Given the historical learning interaction record X of the learner, the learning goal T, and the candidate learning item set D, successively recommend the learning item k_t to the learner, forming a learning path P of length N. During the recommendation process, the interaction record $f_t = \{k_t, score_t\}$ of the learner regarding the learning item k_t at each time step can be observed in real time. The goal is to maximize the knowledge level improvement of the learner on the learning goal T after the learner conducts

learning activities along the recommended learning path.

2.4. RL4ALPR Model

This section presents the overall framework of the RL4ALPR model. Assume that the learning path length is N, meaning that the recommendation is divided into N time steps. Next, let's explain the model details using the t-th time step as an example.

RL4ALPR consists of the following four sub-modules: knowledge level modeling (SAKT), candidate learning item selection (CN), reinforcement learning recommender (A2C), and reward calculation. At each time step, SAKT explores the potential knowledge level of the learner based on their historical interactions; CN selects the candidate learning item set from the prerequisite graph based on the learning items answered by the learner in the previous time step, serving as the action space for A2C; A2C provides the learning items to be answered for the learner; the rewards are passed to A2C to improve the recommendation strategy. The path generation process is modeled as a Markov decision process.

2.4.1. Modeling of Knowledge Level

Given the historical interaction record set $X = \{x_1, x_2, \cdots x_t\}$ for each learner, where the individual interaction $x_i = \{e_i, score_i\}$ represents the learner's response $score_i$ to the learning item or exercise e_i at time point i, with $score_i$ taking values of 1 or 0 indicating whether the learner answered the learning item e_i correctly. The historical interaction record X and the next learning item e_{t+1} are used as the input of the model, and the model's output is the sequence $\widetilde{SCORE} = \left\{\widetilde{score_2}, \widetilde{score_3} \cdots \widetilde{score_{t+1}}\right\}$. The role of the knowledge tracking task is to predict the probability $P(score_{t+1} = 1 \mid e_{t+1}, X)$ that the learner will correctly answer the next learning item. By controlling the next learning item to be answered e_{t+1} in the model input, different $\widetilde{score_{t+1}}$ can be obtained. When the learning items are one-to-one corresponding to the knowledge points, the $\widetilde{score_{(t+1,1)}}$, $\widetilde{score_{(t+1,2)}}$ ... can be concatenated into a vector y, which can represent the learner's implicit knowledge level. The dimension of vector y is determined by the number of learning items.

Based on DKT, SAKT expands an embedding layer to extract historical interaction features, which can identify relevant knowledge points related to the current knowledge point from past interaction records, and assign weights to past answer learning items. Then, it predicts the student's future performance in a specific learning item based on their performance in past learning items. Figure 2 shows the structure of the SAKT module.

The embedding matrices $\hat{M}$ and $\hat{E}$ are input into a self-attention layer based on the scaled dot-product attention mechanism. The role of this layer is to assign a weight to each learning item previously answered, and the weight will affect the correctness of the model's prediction. The scaled dot-product attention mechanism is as follows:

$$Attention(Q, K, V) = soft\max\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{4}$$

Obtain queries and key-value pairs:

$$Q = \hat{E}W^Q, K = \hat{E}W^K, V = \hat{M}W^V \tag{5}$$

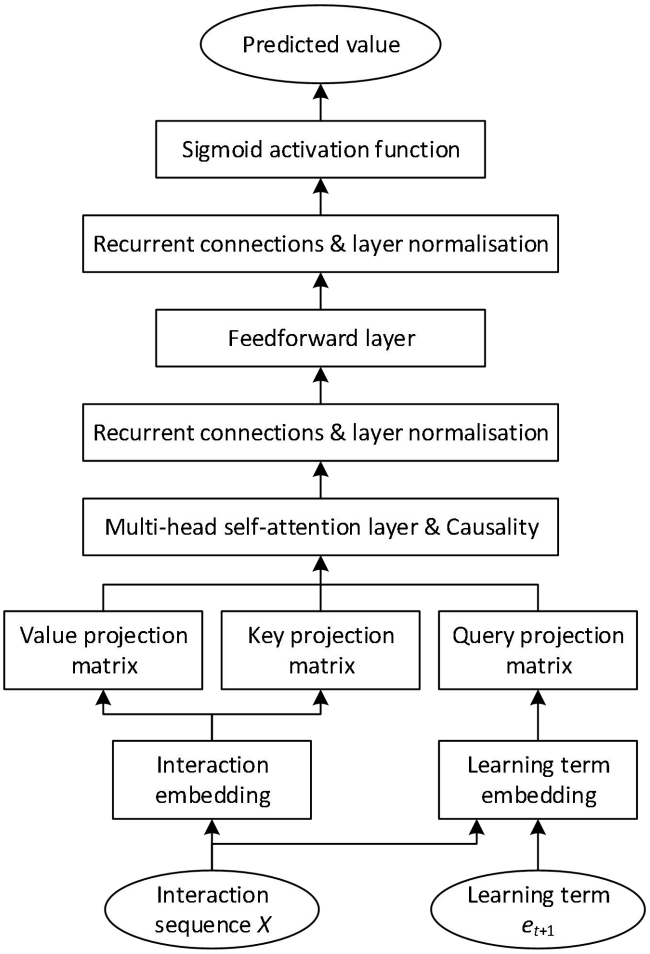

Figure 2. SAKT model structure

W^Q, W^K, and $W^V \in R^{d\times d}$ are the query, key, and value projection matrices, respectively. These projection matrices project the corresponding vectors to different spaces. To be able to handle information representing different spaces simultaneously, a multi-head self-attention mechanism is used for h linear projections:

$$Multihead\left(\hat{M},\hat{E}\right)=Concat\left(head_1,head_2,\cdots head_h\right) \tag{6}$$

Here, $head_i=attention\left(\hat{E}W_i^Q,\hat{M}W_i^K,\hat{M}W_i^V\right)$, and $W^O\in R^{hd\times d}$.

The value $S=Multihead\left(\hat{M},\hat{E}\right)$ obtained from the multi-head self-attention layer is a linear combination. To incorporate non-linearity into the model and consider the interaction between different latent dimensions, a feedforward network is used to obtain the matrix F:

$$F=FFN\left(S\right)=ReLU\left(SW^{(1)}+b^{(1)}\right)W^{(2)}+b^{(2)} \tag{7}$$

Here, $W^{(1)}$ and $W^{(2)}\in R^{d\times d}$, and $b^{(1)}$ and $b^{(2)}\in R^d$ are the parameters during training.

To prevent gradient explosion, gradient vanishing, and to accelerate the convergence of the neural network, layer normalization is used in both the self-attention layer and the feedforward layer.

Finally, the above matrix F is passed through the Sigmoid activation function to predict the probability that students answer the learning items correctly:

$$\widetilde{SCORE}=Sigmoid\left(Fw+b\right) \tag{8}$$

The model trains its parameters by minimizing the cross-entropy loss between the predicted value $\widetilde{score_i}$ and the actual answer $score_i$:

$$L = -\sum_i \left(score_i \log\left(\widetilde{score_i}\right) + \left(1 - score_i\right) \log\left(1 - \widetilde{score_i}\right) \right) \tag{9}$$

2.4.2. Candidate Learning Item Selection

Due to the complex logical relationships among the knowledge points, this paper designs a cognitive navigation algorithm based on a certain central node on the prerequisite condition graph to quickly screen the candidate nodes related to the central node.

Given the prerequisite condition graph G, the learning target T, the central node k_c, and the hop count n, the pseudo-code for using the cognitive navigation algorithm to filter the candidate learning item set D recommended for the agent in the next time step is shown in Algorithm 1:

At each time step, the central node is the knowledge points contained in the learning items recommended to the learner in the previous time step. The time complexity of the cognitive navigation algorithm is related to the size of the prerequisite condition graph G and the hop count n, that is, the time complexity of Algorithm 1 is $O(|G| \cdot n)$.

2.4.3. Recommendation Model Building

In each moment t during the learning path recommendation process of the A2C algorithm, after observing the current state s_t, the agent, using the $\varepsilon - greedy$ strategy based on the policy network $\pi(\cdot | s_t; \theta_t)$, randomly samples an action a_t, which is a learning item k_t. The learner answers k_t, and the environment provides a new state s_{t+1} and a reward r_t, resulting in a trajectory (s_t, k_t, r_t, s_{t+1}) at time t. Then, s_{t+1} will be used as the input for $\pi(\cdot | s_{t+1}; \theta_t)$ to calculate the new action distribution probability, that is, the new probability of each learning item being recommended. Then, based on this probability, a new learning item $\widetilde{k_{t+1}}$ is randomly sampled. $\widetilde{k_{t+1}}$ does not need to be answered by the learner. The sampling of $\widetilde{k_{t+1}}$ is to update the parameters θ and w in the policy network and value network.

After observing the m trajectories $\left\{\left(s_{t+i}, k_{t+i}, r_{t+i}, s_{t+i+1}\right)\right\}_{i=0}^{m-1}$ from time t to $(t+m-1)$, the Multi-step TD Target is calculated first:

$$mTDT_t = \sum_{i=0}^{m-1} \gamma^i \cdot r_{t+i} + \gamma^m \cdot v\left(s_{t+m}; w\right) \tag{10}$$

Here, γ represents the discount rate. Calculate the TD Error:

$$\delta_t = v\left(s_t; w\right) - mTDT_t \tag{11}$$

Update the parameters θ of the policy network $\pi(k_t | s_t; \theta_t)$ using the policy gradient algorithm. Take the derivative of the policy network $\pi(k_t | s_t; \theta_t)$:

$$d_{\theta,t} = \frac{\partial \ln \pi(k_t | s_t; \theta_t)}{\partial \theta_t} \tag{12}$$

Update the parameter θ using gradient ascent:

$$\theta_{t+1} = \theta_t - \beta \cdot \delta_t \cdot d_{\theta,t} \tag{13}$$

Using the TD algorithm with Multi-Step TD Target to update the parameters w of the value network $v(s; w)$, and taking the derivative of the value network $v(s_t; w)$:

$$d_{w,t} = \frac{\partial v(s_t; w)}{\partial w_t} \tag{14}$$

Update parameter w:

$$w_{t+1} = w_t - \alpha \cdot \delta_t \cdot d_{w,t} \tag{15}$$

Among them, $\delta_t \cdot d_{w,t}$ represents the gradient of the loss function, and the loss function is as follows:

$$Loss(w)=\frac{1}{m}\sum_{i=0}^{m-1}\left[v(s_t;w)-mTDT_t\right]^2=\frac{1}{m}\sum_{i=0}^{m-1}\delta_t^2 \tag{16}$$

When updating the policy network, Monte Carlo approximation was used to calculate $g(k_t;\theta)$:

$$\begin{aligned} g(k_t;\theta) &\approx \frac{\partial \ln \pi(k_t \mid s_t;\theta)}{\partial \theta} \\ &\cdot\left[\sum_{i=0}^{m-1}\gamma^i\cdot r_{t+i}+\gamma^m\cdot v(s_{t+m};w)-v(s_t;w)\right] \end{aligned} \tag{17}$$

$$\begin{aligned} -\delta_t &= \sum_{i=0}^{m-1}\gamma^i\cdot r_{t+i}+\gamma^m\cdot v(s_{t+m};w)-v(s_t;w) \\ &= mTDT_t-v(s_t;w) \end{aligned} \tag{18}$$

The reason for choosing $v(s_t;w)$ as the baseline is as follows: $v(s_t;w)$ can be understood as the prediction of the cumulative reward U_t for the future at time t, based on the state s_t, that is, the prediction of the improvement in the learner's knowledge level:

$$v(s_t;w)\approx \mathbb{E}[U_t \mid s_t] \tag{19}$$

It can be used to evaluate the quality of the state s_t at time t. The better s_t is, the larger the value of $v(s_t;w)$ will be. $mTDT_t=\left[\sum_{i=0}^{m-1}\gamma^i\cdot r_{t+i}+\gamma^m\cdot v(s_{t+m};w)\right]$ This can be understood as a prediction of the improvement in the learner's knowledge level based on the states s_t and s_{t+1}:

$$mTDT_t=\sum_{i=0}^{m-1}\gamma^i\cdot r_{t+i}+\gamma^m\cdot v(s_{t+m};w)\approx \mathbb{E}[U_t \mid s_t,s_{t+1}] \tag{20}$$

2.4.4. Reward Calculation

When the recommended learning item is suitable for the current learner and beneficial to the enhancement of their knowledge level, the reward should encourage this behavior; conversely, it should punish this behavior. Therefore, the reward function is defined as the degree of change in the learner's mastery of the target knowledge point after answering the recommended learning item. At each time step during the recommendation process, the reward function is set as follows:

$$r_t=\sum\left(y_{t+1,k}-y_{t,k}\right) \tag{21}$$

Among them, k represents the index of the knowledge point included in the learning objective T, and $y_{t,k}$ represents the mastery level of knowledge point k by the learner at time t.

3. Experiment and Results of Personalized Learning Path Planning

3.1. Performance Testing of the RL4ALPR Model

Programming language: Matlab;
Operating system: Windows 10;
Processor: Intel(R) Core(TM) i5-6200U CPU processor;
Clock speed: 2.40 GHz.

This paper selected six benchmark functions to conduct performance tests and evaluations on the algorithm. F1, F2, and F3 are unimodal functions and have only one optimal solution, which can be used to judge the convergence accuracy of the algorithm; F4, F5, and F6 are multimodal functions and have multiple local optimal solutions, which can be used to judge the algorithm's ability to escape from local optimal solutions. The comparison algorithms are CF-LPR algorithm, DQN-LPR algorithm, and KG-LPR algorithm. Among them, the KG-LPR algorithm improves the binary particle swarm algorithm with a dynamic strategy.

Sphere:

$$F_1(x)=\sum_{i=1}^{n}x_i^2 \tag{22}$$

Step:

$$F_2(x) = \sum_{i=1}^{n} (x_i + 0.5)^2 \tag{23}$$

Rosenbrock:

$$F_3(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right] \tag{24}$$

Rastrigin:

$$F_4(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10n \right] \tag{25}$$

Ackley:

$$F_5(x) = -20\exp\left(-0.2\sqrt{\frac{\sum_{i=1}^{n} x_i^2}{n}} \right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e \tag{26}$$

Griewangk:

$$F_6(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \left[\cos\left(\frac{x_i}{\sqrt{j}} \right) + 1 \right]. \tag{27}$$

The experiment adopts the following two standards to evaluate the performance of the algorithm:

(1) Mean: The average value of the optimal values obtained after 50 runs of the 4 algorithms.

(2) Variance: The variance of the optimal values obtained after 50 runs of the 4 algorithms.

The experimental results of the RL4ALPR model and the three comparison algorithms on six benchmark functions are shown in Table 1. For unimodal functions, there is only one local optimal solution to detect the convergence accuracy.

In the unimodal functions F1, F2, and F3, the mean of the RL4ALPR model is better than that of other algorithms, indicating that this algorithm has higher convergence accuracy than the CF-LPR, DQN-LPR, and KG-LPR algorithms. For multimodal functions, there are multiple local optimal solutions to evaluate whether the algorithm can escape from the local optimal solution. In the multimodal functions F4, F5, and F6, the mean of the RL4ALPR model is significantly better than that of other algorithms, with values of 891.0300, 0.3530, and 0.0539 respectively, indicating that it has stronger ability to escape from the local optimal solution. Overall, the variance of the RL4ALPR model is better than the other algorithms in both unimodal and multimodal functions, indicating that the stability of the algorithm has also been further improved.

Table 1. Experimental results on six benchmark functions

Function	Index	CF-LPR	DQN-LPR	KG-LPR	RL4ALPR
Sphere	Mean	69.3040	58.0320	19.1680	12.6320
	Variance	3.1332	2.7651	2.3738	0.9653
Step	Mean	213.6500	189.6500	137.3500	98.4010
	Variance	5.4341	51641.0000	3.7684	2.6362
Rosenbrock	Mean	9541.5000	8956.4000	3736.6000	2522.1000
	Variance	309.4200	349.7800	278.3600	256.6700
Rastrigin	Mean	897.0400	897.0600	897.0200	891.0300
	Variance	3.3106	3.4865	1.4498	1.4355
Ackley	Mean	1.8317	1.6831	0.9935	0.3530
	Variance	0.0173	0.0166	0.0040	0.0030
Grinwank	Mean	0.3005	0.2564	0.0877	0.0539
	Variance	0.0106	0.0096	0.0074	0.0056

The recommendation results of the learning path may vary due to differences in learners and the number of knowledge points. Therefore, this paper sets up two sets of experiments to verify the performance of the learning path recommendation. The maximum number of iterations is set at 120

times. Experiments 1, 2, and 3 are conducted under the condition that the number of learners remains unchanged, while the number of knowledge points gradually increases; Experiments 4, 5, and 6 are conducted under the condition that the number of knowledge points remains unchanged, while the number of learners gradually increases. The parameter settings are shown in Table 2.

Table 2. Parameter setting

Parameter name	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Quantity of knowledge (M)	30	60	120	60	60	60
Learning resource quantity (N)	6	6	6	6	6	6
Number of learners (K)	6	6	6	12	24	36

The experimental results of the objective function of the adaptive learning path recommendation model RL4ALPR and three comparison algorithms are shown in Table 3. Since the personalized learning path planning problem has a minimization objective function, from an overall perspective, the recommendation method based on the RL4ALPR model achieves the best mean and variance, indicating that this method can recommend more accurate learning paths for learners and has more stable performance. From Experiment 1 to Experiment 3, it can be seen that as the number of knowledge points increases, the mean and variance of the RL4ALPR model in this problem model show a decreasing trend, indicating that its optimization results are more precise; from Experiment 4 to Experiment 6, it can be seen that as the number of learners increases, the mean and variance generated by the RL4ALPR model are also gradually decreasing. This indicates that this method will obtain better results regardless of whether the number of learners or the number of learning resources increases.

Table 3. Test result

Function	Index	CF-LPR	DQN-LPR	KG-LPR	RL4ALPR
Test 1	Avg	65.7668	66.8707	47.1870	43.4057
	Var	153.2598	163.1997	91.3889	43.6661
Test 2	Avg	35.7328	36.3701	25.3490	19.6628
	Var	34.5560	89.6989	30.9377	15.2503
Test 3	Avg	17.0561	16.4458	12.5010	9.9954
	Var	9.4939	21.3058	5.5361	2.8123
Test 4	Avg	303.8500	301.5700	252.4495	195.5802
	Var	795.4795	712.5897	380.8500	114.1503
Test 5	Avg	187.8801	196.6103	149.1896	100.1402
	Var	2619.6998	868.4599	3933.6999	1114.7001
Test 6	Avg	80.5381	83.2793	65.2783	57.0929
	Var	496.1502	166.8901	111.2300	37.3839

3.2. Personalized Learning Path Planning Results

The learning outcome of learners as the learning objective, denoted as LO (hereinafter referred to as target LO), is recorded as E_P after learning through the path, and the calculation method is as shown in Formula (28):

$$E_P = \frac{E_e - E_s}{1 - E_s} \tag{28}$$

This paper takes the learning effect defined by formula (28) as the evaluation index, and takes the probability of the simulated learner answering the questions correctly in the target LO before the path learning as E_s. Subsequently, whenever a LO is learned from the learning path, the simulator will, based on the LO that has been learned in the past, determine the answer result of the questions in that LO, that is, whether it is correct or incorrect. According to this result, the simulator will update the state of the simulated learner and continue to learn new LO, and this process repeats until the path learning is completed. At this point, the probability of answering the questions correctly in the target LO is taken as E_e. Finally, the learning effect E_P of the learning path is calculated by formula (28). In addition, to further verify the efficiency of learning based on the procedural learning path, in the simulation experiment, this paper compares four learning path recommendation methods by using the number of LO required to complete the learning of the target LO and the correct answer rate of the questions in the target LO as evaluation indicators. Among them, the calculation method of the

correct answer rate R_c of the target LO is as shown in formula (29):

$$R_c = \frac{|LP_c|}{|LP_{all}|} \tag{29}$$

Among them, $|LP_{all}|$ represents the total number of learning paths generated in the simulation experiment, and $|LP_c|$ represents the number of learning paths that lead to the correct answers of the target LO questions in the simulation experiment.

This experiment recruited 10 college students as volunteers. All the volunteers had a basic understanding of online learning. Each volunteer chose three courses: C/C++ programming design, Java language programming design, and Python programming design as their learning goals. They used the methods described in this paper and the comparison method to generate learning paths and study. Volunteers were required to select the next unlearned target LO after completing the study of one target LO to ensure that no learning path contained multiple target LO situations. After each learning session, volunteers were required to complete a questionnaire. The results were collected using the 7-point Likert scale method, which included 7 questions, as shown in Table 4.

Table 4. Likert Scale questionnaire

Recommendation process	Q1: The learning path of qbe gives me a clear understanding of the whole learning process.
	Q2: The process of using this learning path can be enough to grasp your learning progress.
Recommended effect	Q3: The learning process of using this learning path suits me very well.
	Q4: The learning path makes it easy for me to master my learning goals.
	Q5. every step of learning path is reasonable.
Path form	Q6: The learning path display is intuitive.
	Q7. learning and using the learning path recommendation system is easy.

In the user experiment section, this chapter collected the questionnaire survey results of the volunteers for each learning path recommendation method, and calculated the average score and standard error of each question. The statistical results are shown in Tables 5 and 6.

The methods in this paper scored higher than the learning path recommendation methods based on graphs, clustering, and processes in terms of the degree of path personalization (Q3), final recommendation effect (Q4), and process recommendation effect (Q5). Among them, the method in this paper scored the highest in the aspect of path personalization (Q3), indicating that this method can more effectively utilize personalized information to recommend suitable learning paths for learners. The method in this paper scored the highest in the aspect of final recommendation effect (Q4), indicating that the recommended learning paths by this method can help learners achieve learning goals more effectively. The method in this paper scored the highest in the aspect of process recommendation effect (Q5), with a score of 4.9, indicating that during the learning process using the learning path, the recommendation model can accurately recommend the optimal learning path based on the learner's immediate knowledge state. Moreover, the standard deviation of the scores of this method in questions Q1 to Q7 was the lowest, indicating that the recommendation effect of this method was the most stable.

In summary, this method can intuitively display the complete learning process, can consider the changes in the learner's knowledge state during the learning process, and dynamically recommend paths. The final recommendation effect is the best among the comparison methods. However, the form of the paths recommended by this method is more complex compared to other methods, and requires learners to have a longer learning and adaptation time.

Table 5. Statistical data of questionnaire results

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
A drawing based learning path recommendation method	4.1	4.2	3.3	3.4	3.5	5.1	5.2
Based on clustering learning path recommendation	4.5	5.0	4.1	4.2	3.5	5.2	5.1
Process learning path recommendation method	2.7	3.2	4.5	5.3	4.5	3.4	4.7
Adaptive learning path recommendation model based on intensive learning	5.5	5.2	5.4	5.3	4.9	5.8	5.6

Table 6. Standard error of each problem

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
A drawing based learning path recommendation method	1.2	0.9	1.5	1.6	1.3	1.1	0.8
Based on clustering learning path recommendation	0.9	0.8	0.9	1.1	1.2	1.1	1.0
Process learning path recommendation method	0.7	0.6	0.4	0.8	0.7	0.5	0.6
Adaptive learning path recommendation model based on intensive learning	0.1	0.2	0.3	0.2	0.1	0.3	0.2

In the simulation experiment section, the average E_P values obtained from the simulation learning based on different path recommendation methods for each course were statistically analyzed. The results of SAKT and DKT are shown in Figures 3 and 4 respectively.

The EP values of the method proposed in this paper in the simulation experiments of different courses are the highest among all methods. Among them, the EP value of the adaptive learning path recommendation model based on reinforcement learning is higher than that of the learning path recommendation method based on graphs and the learning path recommendation method based on clustering, indicating that considering the changes in knowledge states during the learning process and making targeted recommendations can effectively improve the effectiveness of path recommendation.

The learning path recommendation method based on graphs only considers the learning dependencies of LO in the generated path, while ignoring the learning experiences of historical learners in the real environment, resulting in poor path applicability. Therefore, the EP value is lower compared to other methods. The EP value of the clustering-based recommendation method is higher than that of the learning path recommendation method based on graphs, but because it only considers the learning experiences of historical learners and ignores the learning dependencies of LO, the recommended learning path has the problem of an unreasonable learning sequence of LO. The method proposed in this paper achieves better recommendation results.

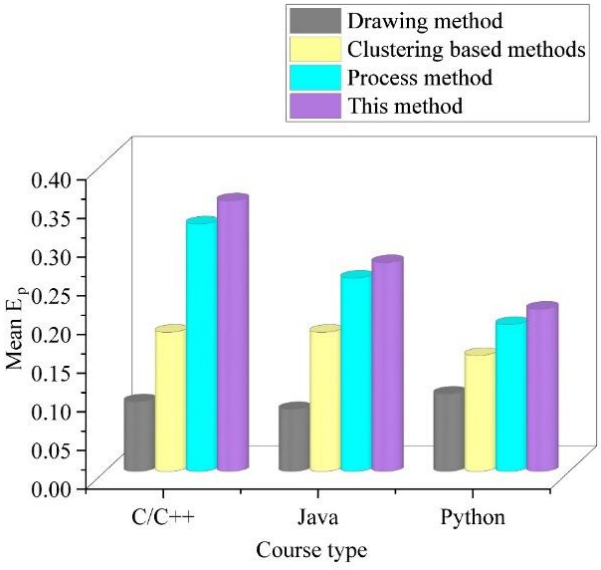

Figure 3. Learning effect of SAKT simulator

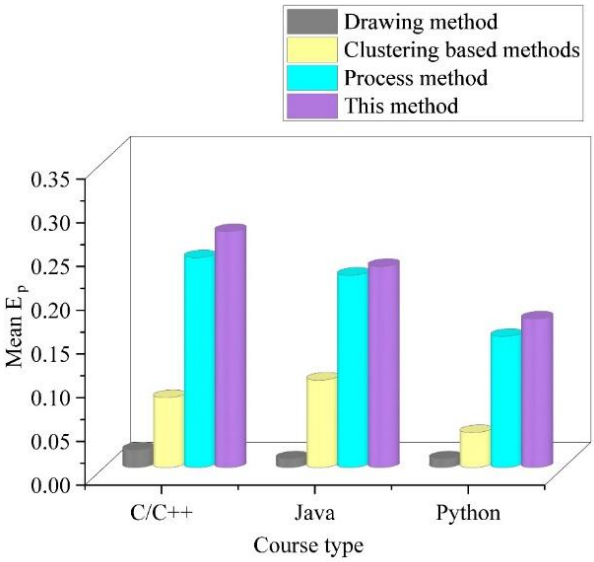

Figure 4. Learning effect of DKT simulator

Figure 5 shows the distribution and average values of the number of learning paths (LO) for different learning paths. The red solid horizontal lines in each part represent the average number of LO. It can be concluded that the average number of LO required for the target LO learning by the method in this paper is less than that of the learning path recommendation methods based on clustering and reinforcement learning, but more than that of the learning path recommendation methods based on graphs. The reason is that the learning path recommendation methods based on graphs generate the shortest path by traversing the knowledge map. Compared with other methods, they do not take into account the actual learning effect of the path. Although the learning path is short, the learning effect is poor. During the experiment, the target LO answer rates of the four learning path recommendation methods calculated were 0.51, 0.65, 0.78, and 0.84 respectively. It can be seen that using the learning path recommended by the method in this paper for learning has the highest target LO answer rate.

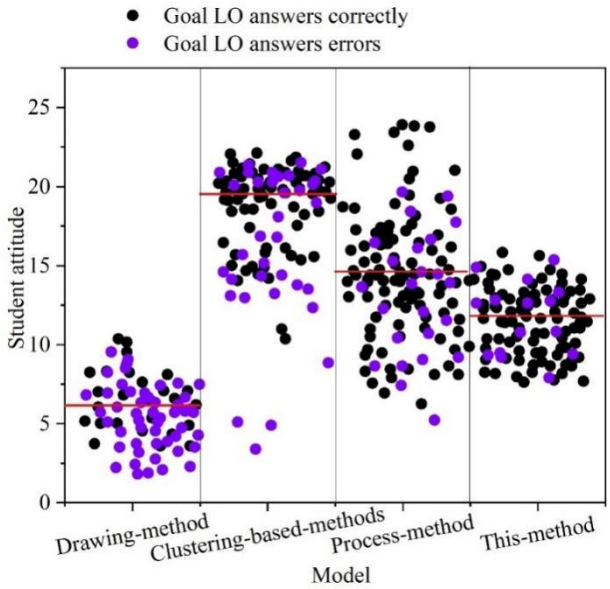

Figure 5. The LO quantity is compared to the mean

4. Career Development Transition Strategies

Employers such as enterprises can post their talent demand information. Through big data analysis and AI algorithm analysis of industry development trends and the talent demand of enterprises, they can generate talent demand reports and transmit them to the educational institutions. The data on the enterprise side is updated dynamically, which is more conducive to educational institutions grasping the

latest developments in the industry and talent demand information, and better cultivating technical and skilled talents suitable for the regional economy.

Based on the talent demand reports transmitted from the enterprise side, educational institutions can condense the knowledge, abilities, and qualities required for each profession in the career positions, and form professional talent training reports. The career development connection model breaks the previous separate educational form. According to the professional talent training reports, the professional development connection of professional talents with a common training goal and training specifications is accurately positioned. The courses are reorganized, designed as a whole, coordinated, and implemented in stages according to the requirements of professional abilities. The educational resources and school advantages are fully utilized, and the corresponding theoretical teaching and practical skills training are strengthened. The segmented training should start from the perspective of exploring the systematic cultivation of technical and skilled talents. On one hand, it focuses on guiding students' learning habits and methods, strengthening basic vocational qualities and practical skills training. On the other hand, it combines actual situations to implement professional teaching standards, reasonably construct a systematic course system, and modularize the course content. This lays the foundation for the formulation and implementation of dynamic and personalized professional talent training plans. In teaching, the credit system is introduced, with course selection as the core. The learning quality of students is measured by grades and credits. This is conducive to optimizing the structure of students' knowledge acquisition, facilitating individualized teaching by teachers, and leveraging students' personal characteristics to cultivate high-quality talents.

From the time students enter the school, a learning file is established in the on-the-job teaching system for them, recording their daily learning habits, learning behaviors, learning effects, etc. The knowledge tracking algorithm models the changing knowledge level of learners. Based on the reinforcement learning adaptive learning path recommendation model, teaching materials for the weak knowledge modules are pushed to them, urging them to strengthen their learning, recording the learning process and conducting tests, analyzing the learning results, generating a phased student learning growth report, analyzing and comparing the students' learning results with the requirements of talent cultivation goals, formulating dynamic personalized professional talent training plans, and intelligently combining course modules. For example, if some students are going to pursue further studies, a training plan conducive to further studies is formulated for them; if some students are directly seeking employment, a training plan to strengthen the skills required for the corresponding position is formulated for them. The personalized professional talent training plan should be dynamically adjusted according to the students' learning situations during implementation, and learning tasks and the modules that need to be strengthened are intelligently pushed.

5. Conclusion

This paper addresses the issue of generating personalized learning paths in online education platforms and proposes an adaptive learning path recommendation model based on reinforcement learning, named RL4ALPR. This model models the learning path recommendation as a Markov decision process, integrates the knowledge tracking algorithm (SAKT) to model the learners' dynamically changing knowledge levels, applies the A2C algorithm of reinforcement learning to recommend learning items, maximizes the learners' benefits, and utilizes the reinforcement learning algorithm to recommend and generate personalized learning paths.

The performance of the RL4ALPR model in single-peak functions F1, F2, F3 and multi-peak functions F4, F5, F6 is superior to the comparison algorithms. The mean values of the RL4ALPR model in multi-peak functions F4, F5, F6 are 891.0300, 0.3530, and 0.0539 respectively. This indicates that this algorithm has better convergence accuracy than the CF-LPR, DQN-LPR, and KG-LPR algorithms, and has stronger ability to escape from local optimal solutions. The variance of the RL4ALPR model in different types of functions is also superior to the comparison algorithms, indicating that the stability of this model is further improved compared to the existing advanced models.

Learning path recommendation methods based on graph learning, clustering learning, process-based learning, and the adaptive learning path recommendation method based on reinforcement learning proposed in this paper, the target LO correct rates of the four learning path recommendation methods are 0.51, 0.65, 0.78, and 0.84 respectively. The learning paths recommended by this method have the highest target LO correct rate for students. This fully demonstrates the effectiveness of the RL4ALPR model in improving students' learning outcomes.

The adaptive learning path recommendation method based on reinforcement learning proposed in this paper has good compatibility with the long-term career development goals of learners. Through the dynamic identification of learning weaknesses and skill gaps using knowledge tracking technology, it

can directly map to the required capabilities for specific career directions, providing a solid data foundation for achieving precise alignment of personalized learning path planning and career development.

References

1. Wu, F., Lu, C., Zhu, M., Chen, H., Zhu, J., Yu, K., ... & Pan, Y. (2020). Towards a new generation of artificial intelligence in China. Nature machine intelligence, 2(6), 312-316.
2. Tang, Y., Liang, J., Hare, R., & Wang, F. Y. (2020). A personalized learning system for parallel intelligent education. IEEE Transactions on Computational Social Systems, 7(2), 352-361.
3. Xu, X., Li, Z., Hin Hong, W. C., Xu, X., & Zhang, Y. (2024). Effects and side effects of personal learning environments and personalized learning in formal education. Education and Information Technologies, 29(15), 20729-20756.
4. Li, W., & Pan, Y. (2023). Image processing-based detection method of learning behavior status of online calssroom students. Physical Communication, 59, 102072.
5. Wang, G., Gong, C., & Wang, S. (2022, June). A review of automatic detection of learner states in four typical learning scenarios. In International Conference on Human-Computer Interaction (pp. 53-72). Cham: Springer International Publishing.
6. Raj, N. S., & Renumol, V. G. (2024). An improved adaptive learning path recommendation model driven by real-time learning analytics. Journal of Computers in Education, 11(1), 121-148.
7. Fiqri, M., & Nurjanah, D. (2017, April). Graph-based domain model for adaptive learning path recommendation. In 2017 IEEE global engineering education conference (EDUCON) (pp. 375-380). IEEE.
8. Nadimpalli, V. K., Maier, R., Ezer, T., Bugert, F., Staufer, S., Röhrl, S., ... & Mottok, J. (2025, June). Nestor: A personalized learning path recommendation algorithm for adaptive learning environments. In Proceedings of the 6th European Conference on Software Engineering Education (pp. 49-59).
9. Vagale, V., Niedrite, L., & Ignatjeva, S. (2020). Application of the Recommended Learning Path in the Personalized Adaptive E-learning System. Baltic Journal of Modern Computing, 8(4).
10. Ma, Y., Wang, L., Zhang, J., Liu, F., & Jiang, Q. (2023). A personalized learning path recommendation method incorporating multi-algorithm. Applied Sciences, 13(10), 5946.
11. MASHWANI, W. K., ALZAHRANI, A., & ALZAHRANI, A. O. (2023). Smart E-Learning Framework for Personalized Adaptive Learning and Sequential Path Recommendations Using Reinforcement Learning. IEEE Access, Hoes Lane, Piscataway, USA, Tech. Rep.
12. Chen, J. Y., Saeedvand, S., & Lai, I. W. (2023). Adaptive learning path navigation based on knowledge tracing and reinforcement learning. arXiv preprint arXiv:2305.04475.
13. Yun, Y., Dai, H., An, R., Zhang, Y., & Shang, X. (2024). Doubly constrained offline reinforcement learning for learning path recommendation. Knowledge-Based Systems, 284, 111242.
14. Haldar, S., Sengupta, S., & Das, A. K. (2025). Personalized Learning Path Recommendation using Graph Reinforcement Learning. Procedia Computer Science, 258, 3480-3489.
15. Wan, H., Che, B., Luo, H., & Luo, X. (2023, July). Learning path recommendation based on knowledge tracing and reinforcement learning. In 2023 IEEE International Conference on Advanced Learning Technologies (ICALT) (pp. 55-57). IEEE.
16. Stephen, J. S. (2024). Career planning, professional development, and lifelong learning. In Academic Success in Online Programs: A Resource for College Students (pp. 199-212). Cham: Springer Nature Switzerland.
17. Bridgstock, R., Grant-Iramu, M., & McAlpine, A. (2019). Integrating career development learning into the curriculum: Collaboration with the careers service for employability. Journal of Teaching and Learning for Graduate Employability, 10(1), 56-72.

18. Beausaert, S., Segers, M., & Grohnert, T. (2014). Personal development plan, career development, and training. The Wiley Blackwell handbook of the psychology of training, development, and performance improvement, 336-353.
19. Dean, B. A., Ryan, S., Glover-Chambers, T., West, C., Eady, M. J., Yanamandram, V., ... & O'Donnell, N. (2022). Career development learning in the curriculum: what is an academic's role?. Journal of Teaching and Learning for Graduate Employability, 13(1), 142-154.

About the Author
Qing Xiong, female (born September 1981), Han ethnicity, from Wuhan, Hubei Province, holds a doctoral degree from the National University of Technology Malaysia (currently pursuing studies) and a master's degree from the Technical University of Dresden, Germany. She is a senior lecturer in educational management with research focuses on vocational education curriculum development, application of vocational teaching methodologies, and localization of German vocational education models.

AZLAN BIN ABDUL LATIB, Doctor of Philosophy, Universiti Polytechnics Malaysia, Associate Professor Research direction: Technical and Vocational Education